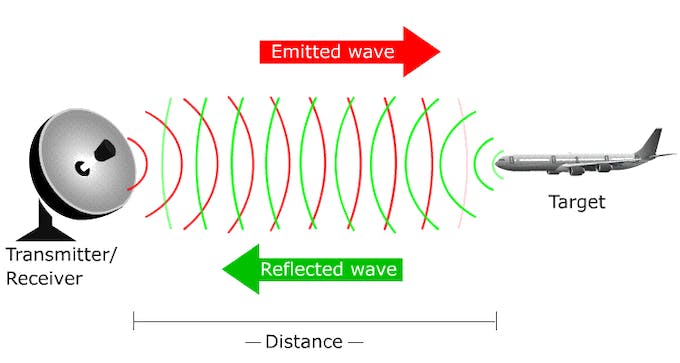
**Mehran University of Engineering and Technology**

**THE RADAR SYSTEM**

[](javascript:openLightBox('34b7676bca',%200);)

**2**

**0**

**2**

**2**

**Insharah Sultana (20CS028)**

**Haseebullah Channa (20CS010)**

**Submitted to:**

**Ma’am Bushra Naz**

**Adeena Khanzada (20CS068)**

**Certificate of Approval**

The undersigned certify that they have read and recommended to the Department of Computer System for acceptance, a project report entitled “RETRO CAR GAME” submitted by:

1. Insharah sultana (20CS028)
2. Haseebullah channa(20CS010)

in partial fulfillment of the requirement for the Project of Computer Graphics of Third semester of Bachelor of Computer System.

**TABLE OF**

# CONTENTS

|  |  |
| --- | --- |
| 1. Abstract 2. Introduction | 1. Conclusion 2. Acknowlegment |

1. Background History
2. Features
3. Components of C++
4. Source Code
5. Output
6. Requirements

# ABSTRACTS

The project “The Radar System” is based on simple hardware as well as software. The components are very simple and handy, which can easily demonstrate the concept of the radar system used widely at military bases, in civil aviation departments, and in other fields.

The Arduino is coded very simply so that it can be easily understood by the users and can be easily edited accordingly with the corresponding hardware. The case study discusses the use of simpler components to build the abstraction. In addition, this report will also discuss the combination of simpler components with the greatest output.

# INTRODUCTION

## RADAR SYSTEM

The word RADAR stands for Radio Detection and Ranging. Radar is an object detection system that uses microwaves to determine the range, height, direction, and speed of objects within about 100 miles of their location.

A radar antenna transmits radio waves or microwaves that bounce off anything in its path. Because of this, we can easily determine the object in the radar range.

# BACKGROUND HISTORY

The history of radar (where radar stands for radio detection and ranging) started with experiments by Heinrich Hertz in the late 19th century that showed that radio waves were reflected by metallic objects. This possibility was suggested in James Clerk Maxwell's seminal work on electromagnetism. However, it was not until the early 20th century that systems able to use these principles were becoming widely available, and it was German inventor Christian Hülsmeyer who first used them to build a simple ship detection device intended to help avoid collisions in fog (Reichspatent Nr. 165546). True radar, such as the British Chain Home early warning system provided directional information to objects over short ranges, was developed over the next two decades. The development of systems able to produce short pulses of radio energy was the key advance that allowed modern radar systems to come into existence.

# CEP (Complex Engineering Problem):

How to create the radar system for military bases?

**SOLUTION:** The simple radar system can be created as a military base radar system by decreasing its delay to increase its speed to rapidly show any moment and body passing near the base.

# COMPONENTS

These are the different components that are involved to build the successful resultant.

**Ultrasonic sensor:**

An ultrasonic sensor is a proximity sensor used to measure the distance of an object or object. It transmits ultrasonic waves to detect objects and converts the reflected waves into electrical signals. These sound waves travel faster than the speed of sound that humans can hear. It has two main components: the transmitter & receiver. The transmitter emits the sound using a piezoelectric crystal, and the receiver encounters the sound after it has travelled to and from the target.

## FORMULA:

The formula for this calculation is,

D = ½ T x C

Where,

* D = distance,
* T = time
* C = speed of sound which is 343 meters/second.

## Servo Motor:

The **servo motor**is a simple DC motor that can be controlled for specific angular rotation with the help of additional servomechanism. This motor will only rotate as much as we want and then stop. The servo motor is a closed-loop mechanism that uses positional feedback to control the speed and position. The servo motor is unlike a standard electric motor which starts and stops according to the power input. According to the signal, the servo motor will work.

Nowadays, servo motors are widely used in industrial and robotics applications. They are also commonly seen in remote-controlled toy cars, RC planes, and CD or DVD players. Besides these, we see hundreds of applications in our daily life that use a servo motor. To know more about the servo motor.

## Arduino:

The name Arduino comes from a bar in Ivrea, Italy, where some of the founders of the project used to meet. The bar was named after Arduin of Ivrea, who was the margrave of the March of Ivrea and King of Italy from 1002 to 1014. Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards can read inputs (lights from sensors, fingers on buttons, or Twitter messages) and turn them into outputs (activate motors, turn on LEDs, post online). It consists of a programmable circuit board (called a microcontroller) and off-the-shelf software called the Arduino Integrated Development Environment (IDE) that is used to write computer code and upload it to the physical board.

## Bread Board:

A breadboard is a simple device designed to create circuits without soldering.

The breadboard is the bread and butter of DIY electronics. Breadboards allow beginners to get acquainted with circuits without the need for soldering, and even seasoned tinkerers use breadboards as starting points for large-scale projects. The two larger pieces of wire down each side are typically used to connect a power source to the board. They are usually referred to as **power rails**. Another small piece of wire running vertically across the board is used for the components in the circuit. This diagram will help you visualize this pattern from above.

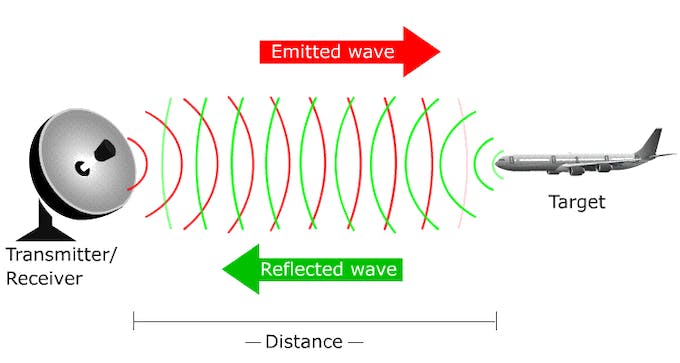
## Working:

Radio waves or microwaves are radiated out from the radar into free space. Some of these waves are intercepted by reflecting objects.

These intercepted radio waves hit the target and are reflected in many different directions. Some of these waves can be directed back toward the radar, where they are received and amplified.

If these waves are received again at their origin, then it means an object is in the propagation direction.

The modern radar system is very advanced and used in highly diverse applications such as Air traffic control, Air-defense system, radar Astronomy, Antimissile system, Outer space Surveillance system, and many more.

[](javascript:openLightBox('34b7676bca',%200);)

**C A R C O M P A R I S O N S Y S T E M**

# SOURCE CODE

## Arduino code:

// Includes the Servo library

#include <Servo.h>.

// Defines Tirg and Echo pins of the Ultrasonic Sensor

const int trigPin = 10;

const int echoPin = 11;

// Variables for the duration and the distance

long duration;

int distance;

Servo myServo; // Creates a servo object for controlling the servo motor

void setup() {

pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output

pinMode(echoPin, INPUT); // Sets the echoPin as an Input

Serial.begin(9600);

myServo.attach(12); // Defines on which pin is the servo motor attached

}

void loop() {

// rotates the servo motor from 15 to 165 degrees

for(int i=15;i<=165;i++){

myServo.write(i);

delay(30);

distance = calculateDistance();// Calls a function for calculating the distance measured by the Ultrasonic sensor for each degree

Serial.print(i); // Sends the current degree into the Serial Port

Serial.print(","); // Sends addition character right next to the previous value needed later in the Processing IDE for indexing

Serial.print(distance); // Sends the distance value into the Serial Port

Serial.print("."); // Sends addition character right next to the previous value needed later in the Processing IDE for indexing

}

// Repeats the previous lines from 165 to 15 degrees

for(int i=165;i>15;i--){

myServo.write(i);

delay(30);

distance = calculateDistance();

Serial.print(i);

Serial.print(",");

Serial.print(distance);

Serial.print(".");

}

}

// Function for calculating the distance measured by the Ultrasonic sensor

int calculateDistance(){

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

duration = pulseIn(echoPin, HIGH); // Reads the echoPin, returns the sound wave travel time in microseconds

distance= duration\*0.034/2;

return distance;

}

## Processing code:

import processing.serial.\*; // imports library for serial communication

import java.awt.event.KeyEvent; // imports library for reading the data from the serial port

import java.io.IOException;

Serial myPort; // defines Object Serial

// defubes variables

String angle="";

String distance="";

String data="";

String noObject;

float pixsDistance;

int iAngle, iDistance;

int index1=0;

int index2=0;

PFont orcFont;

void setup() {

size (1200, 700); // \*\*\*CHANGE THIS TO YOUR SCREEN RESOLUTION\*\*\*

smooth();

myPort = new Serial(this,"COM5", 9600); // starts the serial communication

myPort.bufferUntil('.'); // reads the data from the serial port up to the character '.'. So actually it reads this: angle,distance.

}

void draw() {

fill(98,245,31);

// simulating motion blur and slow fade of the moving line

noStroke();

fill(0,4);

rect(0, 0, width, height-height\*0.065);

fill(98,245,31); // green color

// calls the functions for drawing the radar

drawRadar();

drawLine();

drawObject();

drawText();

}

void serialEvent (Serial myPort) { // starts reading data from the Serial Port

// reads the data from the Serial Port up to the character '.' and puts it into the String variable "data".

data = myPort.readStringUntil('.');

data = data.substring(0,data.length()-1);

index1 = data.indexOf(","); // find the character ',' and puts it into the variable "index1"

angle= data.substring(0, index1); // read the data from position "0" to position of the variable index1 or thats the value of the angle the Arduino Board sent into the Serial Port

distance= data.substring(index1+1, data.length()); // read the data from position "index1" to the end of the data pr thats the value of the distance

// converts the String variables into Integer

iAngle = int(angle);

iDistance = int(distance);

}

void drawRadar() {

pushMatrix();

translate(width/2,height-height\*0.074); // moves the starting coordinats to new location

noFill();

strokeWeight(2);

stroke(98,245,31);

// draws the arc lines

arc(0,0,(width-width\*0.0625),(width-width\*0.0625),PI,TWO\_PI);

arc(0,0,(width-width\*0.27),(width-width\*0.27),PI,TWO\_PI);

arc(0,0,(width-width\*0.479),(width-width\*0.479),PI,TWO\_PI);

arc(0,0,(width-width\*0.687),(width-width\*0.687),PI,TWO\_PI);

// draws the angle lines

line(-width/2,0,width/2,0);

line(0,0,(-width/2)\*cos(radians(30)),(-width/2)\*sin(radians(30)));

line(0,0,(-width/2)\*cos(radians(60)),(-width/2)\*sin(radians(60)));

line(0,0,(-width/2)\*cos(radians(90)),(-width/2)\*sin(radians(90)));

line(0,0,(-width/2)\*cos(radians(120)),(-width/2)\*sin(radians(120)));

line(0,0,(-width/2)\*cos(radians(150)),(-width/2)\*sin(radians(150)));

line((-width/2)\*cos(radians(30)),0,width/2,0);

popMatrix();

}

void drawObject() {

pushMatrix();

translate(width/2,height-height\*0.074); // moves the starting coordinats to new location

strokeWeight(9);

stroke(255,10,10); // red color

pixsDistance = iDistance\*((height-height\*0.1666)\*0.025); // covers the distance from the sensor from cm to pixels

// limiting the range to 40 cms

if(iDistance<40){

// draws the object according to the angle and the distance

line(pixsDistance\*cos(radians(iAngle)),-pixsDistance\*sin(radians(iAngle)),(width-width\*0.505)\*cos(radians(iAngle)),-(width-width\*0.505)\*sin(radians(iAngle)));

}

popMatrix();

}

void drawLine() {

pushMatrix();

strokeWeight(9);

stroke(30,250,60);

translate(width/2,height-height\*0.074); // moves the starting coordinats to new location

line(0,0,(height-height\*0.12)\*cos(radians(iAngle)),-(height-height\*0.12)\*sin(radians(iAngle))); // draws the line according to the angle

popMatrix();

}

void drawText() { // draws the texts on the screen

pushMatrix();

if(iDistance>40) {

noObject = "Out of Range";

}

else {

noObject = "In Range";

}

fill(0,0,0);

noStroke();

rect(0, height-height\*0.0648, width, height);

fill(98,245,31);

textSize(25);

text("10cm",width-width\*0.3854,height-height\*0.0833);

text("20cm",width-width\*0.281,height-height\*0.0833);

text("30cm",width-width\*0.177,height-height\*0.0833);

text("40cm",width-width\*0.0729,height-height\*0.0833);

textSize(40);

text("HIA Project ", width-width\*0.875, height-height\*0.0277);

text("Angle: " + iAngle +" °", width-width\*0.48, height-height\*0.0277);

text("Distance: ", width-width\*0.26, height-height\*0.0277);

if(iDistance<40) {

text(" " + iDistance +" cm", width-width\*0.225, height-height\*0.0277);

}

textSize(25);

fill(98,245,60);

translate((width-width\*0.4994)+width/2\*cos(radians(30)),(height-height\*0.0907)-width/2\*sin(radians(30)));

rotate(-radians(-60));

text("30°",0,0);

resetMatrix();

translate((width-width\*0.503)+width/2\*cos(radians(60)),(height-height\*0.0888)-width/2\*sin(radians(60)));

rotate(-radians(-30));

text("60°",0,0);

resetMatrix();

translate((width-width\*0.507)+width/2\*cos(radians(90)),(height-height\*0.0833)-width/2\*sin(radians(90)));

rotate(radians(0));

text("90°",0,0);

resetMatrix();

translate(width-width\*0.513+width/2\*cos(radians(120)),(height-height\*0.07129)-width/2\*sin(radians(120)));

rotate(radians(-30));

text("120°",0,0);

resetMatrix();

translate((width-width\*0.5104)+width/2\*cos(radians(150)),(height-height\*0.0574)-width/2\*sin(radians(150)));

rotate(radians(-60));

text("150°",0,0);

popMatrix();

}

# OUTPUT

# REQUIREMENTS

## HARDWARE REQUIRED

* Arduino UNO
* Servo motor
* Ultra Sonic sensor
* Bread board
* wires
* **Printer:** to print the required documents of the project
* **Compact Drive**
* **Processor:** Intel(R) Core(TM) **RAM:** 512 MB or more than 512MB
* **Hard Disk:** 80 GB or more than 80GB.

## SOFTWARE REQUIRED

* **Operating system:** Windows 10
* **Turbo C++,** for execution of program and **Ms word,** for presentation of output. **Compiler:** we have used DEV C++ to write source code but other compilers also used which includes code block, MinGW, Borland C++ and etc.
* Processing 3 installed
* Arduino software

# WORKING:

Radio waves or microwaves are emitted from radar into free space. Some of these waves are blocked by reflective objects.

This intercepted radio wave hits the target and is reflected in various directions. Some of these waves can return to the radar to be received and amplified.

If this wave is received back from the origin, it means that the object is in the direction of propagation. The output is shown on the screen as the ultrasonic sensor detect the object so the screen shows red and when there is no object detected it shows light.

Modern radar systems are very advanced and used in a wide variety of applications such as air traffic control, air defense systems, radar astronomy, missile defense systems, and space surveillance systems.

# CONCLUSION

This Radar System is only between the range of 35-40cm. It can also be created in higher ranges and can be used in military bases with wider ranges and speedy response of receiver and transmitter. The simplest circuit with the most simplest source code with the achievement of successful result will help in other scenario too.

We take this opportunity to thank all our lecturers who have

directly or indirectly helped our project.

We pay our respects and love to our parents and all other family

members and friends for their love and encouragement

throughout our career. Last but not the least we express our

thanks to our friends for their cooperation and support.

**We thank you Sir Rizwan Badar Baloch**

**for Preparing our Future by teaching us**

**Object Oriented Programming.**

**Contact**

haseebchanna028@yahoo.com

insharahsultana@gmail.com

**ACKNOWLEDGEMENTS**